

II. Applicant's Response to the Examiner's Rejections

The Applicant traverses the aforementioned claim rejections for the reasons set forth in greater detail below.

A. Improper Finality of Rejection

According to the Office Action, the previous amendment to the claims did not add any new claim limitations. (Office Action, page 2, ref # 4). The Office Action also stated a new ground of rejection based on Favor. (Office Action, page 4, ref # 8). However, the Office Action asserts that the Applicant's amendment necessitated the new ground of rejection and consequently the rejection is made final citing MPEP §706.07(a). (Office Action, page 4, ref # 10). Since the Office Action acknowledges that no new claims limitations were added, the Applicants respectfully disagree that the Applicant's amendments necessitated a new ground of rejection. Therefore, since the Office Action introduced a new ground of rejection not believed necessitated by the Applicant's amendments, the finality of the rejection is improper and should be removed.

B. 35 U.S.C. §103(a)

According to the Office Action, claims 1-6, 8-13, 15, and 17-18 are rejected under 35 U.S.C. §103(a) over Guttag et al. "Guttag" U.S. Patent No. 6,173, 394 in view of applicant's specification.

Firstly, the Office Action fails to establish a prima facie case of Obviousness by failing to show how each and every element in the claims is taught by the reference. To establish a prima facie case of obviousness, each and every element arranged as required by the claim must be taught or suggested in the prior art. MPEP 2143.03. Therefore, a general reference to several pages in a reference does not clearly explain with specificity how each claim is rejected. MPEP

706, CFR 1.104(c)(2). If the Examiner maintains this rejection, Applicants respectfully request a recitation in Guttag of each and every element as claimed.

GUTTAG FAILS AT LEAST TO TEACH "EMULATING NON-NATIVE INSTRUCTIONS USING NATIVE INSTRUCTIONS CONTAINING THE FLAG MODIFICATION ENABLE BIT"

Secondly, rather than teaching "emulating non-native instructions using native instructions containing the flag modification enable bit", Guttag is directed to the execution of a conditional store instruction, and as a result, is directed to a problem different than the claims.

(Guttag, Col. 6, lines 29-35). Similarly, Guttag, as cited in the Office Action, teaches:

The "N C V Z" field (bits 28-25) indicates which bits of the status are protected from alteration during execution of the instruction. The *conditions* of the status register are: N negative; C carry; V overflow; and Z zero. If one or more of these bits are set to "1", the corresponding *condition* bit or bits in the status register are protected from modification during execution of the instruction. Otherwise the status bits of status register 210 are set normally according to the resultant of arithmetic logic unit 230.

(Guttag, Col. 131, lines 43-51). (emphasis added). As understood, Guttag is directed to the execution of conditional instructions, and as such does not teach "emulating non-native instructions using native instructions containing the flag modification enable bit." Further, the Office Action acknowledges that Guttag does not teach "emulating non-native instructions using native instructions containing the flag modification enable bit" by alleging that these elements as arranged in the claims are taught by the applicant's disclosure. Therefore, since Guttag fails to teach each and every element as arranged in the claims, the Office Action fails to establish a *prima facie* case of obviousness.

Thirdly, Guttag teaches an approach opposite from the claims because rather than "emulating non-native instructions using native instructions containing the flag modification

enable bit", Guttag teaches performing all accesses to memory in a single cycle using a RISC instruction. "Master processor 60 is preferably a 32 bit reduced instruction set computer (RISC) processor including a hardware floating point calculation unit. According to the RISC architecture, all accesses to memory are performed with load and store instructions and most integer and logical operations are performed on registers in a single cycle." (Guttag, Col. 12 line 65 - Col. 13 line 3). Since all accesses to memory are performed with RISC instructions rather than non-native instructions, Guttag does not teach "emulating non-native instructions using native instructions containing the flag modification enable bit."

**GUTTAG DOES NOT TEACH NON-NATIVE INSTRUCTIONS CONTAINING
THE FLAG MODIFICATION ENABLE BIT**

Fourthly, since Guttag does not teach, as acknowledged in the Office Action, "emulating non-native instructions using native instructions", Guttag does not teach non-native instructions containing the flag modification enable bit. As stated in the previous response, there is no disclosure in Guttag or applicants specification regarding emulating non-native instructions using native instructions containing the flag modification enable bit. Therefore, since Guttag fails to teach each and every element as arranged in the claims, the Office Action fails to establish a prima facie case of obviousness.

**THE OFFICE ACTION FAILS TO PROVIDE ANY MOTIVATION TO
COMBINE THE REFERENCES**

Sixthly, According to the Office Action, the specification teaches that there was a need to decouple the handling of the flags from the instruction type. However, this mischaracterizes the specification because the specification states "[o]ne problem may be to save and restore flag states." As a result, the Office Action fails to provide any motivation to combine the reference as cited and as a result, the Office Action fails to establish a prima facie case of obviousness.

Seventhly, the "save and restore" method cited in the specification teaches away from Guttag because Guttag as cited teaches an approach for the protection of flags as asserted by the Office Action. As a result, since the references teach away from each other, the combination of the two references as cited is improper.

Eighthly, the teaching or suggestion to make the claimed combination may not be based on the Applicant's disclosure. MPEP 2142. Since the Office Action cites the Applicant's disclosure to provide such motivation, a prima facie case of obviousness is not established.

C. 35 U.S.C. §102(b)

According to the Office Action, claims 1-6, 10-13, and 15 are rejected under 35 U.S.C. §102(b) over Favor "Favor" (WO 97/13194). The Office Action merely recites "The examiner would recommend that applicants read page 35, line 1 through page 38, line 11 at a minimum, before responding."

Firstly, the Office Action fails to establish a prima facie case of anticipation by failing to show how each and every element in the claims is taught by the reference. A claim is anticipated only if each and every element arranged as required by the claim is found in a single prior art reference. MPEP 2131. Therefore, a general reference to several pages in a reference does not clearly explain with specificity how each claim is rejected. MPEP 706, CFR 1.104(c)(2). If the Examiner maintains this rejections, Applicants respectfully request a recitation of each and every element as claimed in Favor.

FAVOR FAILS TO TEACH "DETERMINING WHETHER THE AT LEAST ONE FLAG MODIFICATION ENABLE BIT ALLOWS UPDATING OF AT LEAST ONE FLAG IN RESPONSE TO EXECUTING THE OPERATIONAL CODE"

Secondly, Favor fails to teach "determining whether the at least one flag modification enable bit allows *updating of at least one flag in response to executing the operational code.*" In contrast to the claims, the cited portions of Favor teaches: "

Decoupling of condition code handling from operation type, using the independent TYPE 612 and set status (SS) field 624, allows some operations to be defined which **do not update the flags.** Accordingly, for those circumstances in which updating of condition flags is not necessary, it is highly advantageous to disable flag updating to avoid unnecessary dependency on previous flag values.

(Favor, Page 38, lines 7-10). (emphasis added). The cited portion of Favor teaches that some operations do not update the flags, whereas the claims require an opposite approach, updating of at least one flag in response to executing the operational code. For at least these reasons, Favor fails to teach each and every element as arranged in the claims. Consequently, Favor does not anticipate the claims and as a result this rejection is improper.

Thirdly, Favor teaches decoupling of condition code handling from operation rather than "determining whether the at least one flag modification enable bit allows updating of at least one flag in response to executing the operational code." For at least these reasons, Favor fails to teach each and every element as arranged in the claims. Consequently, Favor does not anticipate the claims and as a result this rejection is improper.

In addition, claim 5 requires, for example, providing a variable length instruction emulator that uses fixed length native instructions to emulate variable link instructions in evaluating the flag modification enabled bit to preserve flag bit settings for variable length instructions that are emulated using the fixed blank native instructions. Applicants again reassert

the relevant remarks made with respect to the failure to show adequate motivation and also note that the Guttag reference is not directed to a variable length instruction emulation circuit or method. Again it appears that the motivation comes from applicants' own invention. As previously stated, Favor fails to teach "determining whether the at least one flag modification enable bit allows updating of at least one flag in response to executing the operational code."

As to claims 2, 3, 4, 6, 13, 15 and 17, Applicants at least reassert that the references do not teach each and every element as arranged in the claims with respect to claim 1.

With respect to claims 9 and 18, applicants note that these claims require, among other things, converting received variable length X86 instructions to a plurality of native instructions when the native instructions have a flag modification enable bit set to allow changing of the non-native instruction flags in response to execution of the native instructions. Again, Guttag is silent as to any instruction conversion. Moreover, the claim requires, that as to unconvertible instructions, the non-native instruction emulator emulates unconverted variable length instructions that include flag modification enable bits set to prevent changing of non-native instruction flags in response to the execution of the native instructions for the unconverted variable length instruction. A setting of the modification enable bits based on conversion and unconvertible instructions is not taught or suggested by the combination of teachings.

Applicant respectfully submits that the claims are in condition for allowance and respectfully request that a timely Notice of Allowance be issued in this case. The Examiner is invited to contact the below listed attorney if the Examiner believes that a telephone conference will advance the prosecution of this application.

Respectfully submitted,

Date: March 17, 2003

Vedder, Price, Kaufman & Kammholz
222 North LaSalle Street
Chicago, Illinois 60601
PHONE: (312) 609-7970
FAX: (312) 609-5005

By


Themi Anagnos, Reg. No. 47,388

RECEIVED
CENTRAL FAX CENTER

SEP 30 2003

OFFICIAL